Lab #1

Student Name: Cem Emir Senyurt

Student #: 45064076

Date: 01/16/2022

1. OBJECTIVE

* 1 – Bit Full Adder:

With this I derived 1-bit Full Adder. We have 3 inputs: A, B, Cin ; 2 Outputs: Sum, Cout. From Full adder truth table, I could be able to find Product of sum equations for the sum and Cout but they were already given. Then, I assign the equations to outputs with bitwise operators.

* 4 – Bit Full Adder:

With this I derived 4 – bit Full Adder. Again we have 3 inputs: A, B, Cin ; 2 Outputs: Sum, Cout. However, this time we are dealing with 4 bits of operation. However, the 4-bits of operations are actually 4 of 1 full bit adder. That’s why I used the same module of Full Adder, added that as a source file and derived 4 different 1-bit full adder side by side as I draw below. It did worke and it was quite fun!

* Multiplexer 2:1

Multiplexer is a signal-based tool that we used. Multiplexer 2:1 means that we have 2 inputs (without the signal), and 1 output. Basically, we have 3 inputs: D1, D2, S. One of them, which is S, is the Signal, and one output: Y. Depending on what the Signal is (in our case 0 or 1), the output is connected to the input that Signal chose. That’s why I wrote a case function which enables my code to determine the signal is 0, then make the output Y = D1, or the signal is 1, then make the output Y = D2.

* Multiplexer 4:1

Multiplexer 4:1 means that we have 4 inputs (without the signal), and 1 output. This time the signal can be (in decimal) 00, 01, 02, 03 which makes the output equals, in order, the inputs: D1, D2, D3, D4. That’s why I wrote a case statement where I declared the all the cases available for the output.

1. PROCEDURE

* 1 – Bit Full Adder:

Diagram, letter

Description automatically generated

* 4 – Bit Full Adder:

A picture containing text, whiteboard

Description automatically generated

* Multiplexer 2:1

Diagram, schematic

Description automatically generated

* Multiplexer 4:1

Diagram

Description automatically generated

1. SIMULATION RESULTS

* 1- Bit Full Adder

Between 0-20 ns, values are in the screenshot as expected.

A screenshot of a computer

Description automatically generated

Between 20-40 ns values are in the screenshot as expected (given).

Chart

Description automatically generated with medium confidence

Between 40-60 ns, values are in the screenshot as expected and calculated.

A screenshot of a computer

Description automatically generated

* 4 – Bit Full Adder

Chart

Description automatically generatedThe a, b actually the changing values of our input a, b. As I stated above those inputs are vectors, so we have more than 1 variable.

I am putting one more screen shot and click the inputs to be able to show variables of the vector:

Graphical user interface

Description automatically generated with medium confidence

* Multiplexer 2:1

Between 0-20 ns:

A picture containing timeline

Description automatically generated

Between 20-40 ns:

A screenshot of a computer

Description automatically generated

* Multiplexer 4:1

Between 0-20 ns:

Graphical user interface, table

Description automatically generated

Between 20-40 ns:

A screenshot of a computer

Description automatically generated

Between 40-60 ns:

Graphical user interface

Description automatically generated

Between 60-80 ns

Chart

Description automatically generated with low confidence